



myMed: an ad hoc Framework for Executing Homogeneous Open Social Networks

Luigi Liquori

► To cite this version:

Luigi Liquori. myMed: an ad hoc Framework for Executing Homogeneous Open Social Networks. 2014. hal-01146061

HAL Id: hal-01146061

<https://inria.hal.science/hal-01146061>

Preprint submitted on 27 Apr 2015

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

myMed: an *ad hoc* Framework for Executing Homogeneous Open Social Networks

Luigi Liquori

Abstract myMed is an experimental framework for implementing and deploying, on the top of a built-in cloud platform, many Open Social Networks (OSN) that could take advantage of sharing common software modules, hardware resources, making inter-communication and inter-interaction simpler and improving rapid development. This paper discuss the main advantages in coordinating and communicating different OSN built and hosted on the same framework and platform.

1 Introduction

The explosion of different Open Social Networks (OSN) “running” in the internet arena has changed the habits of mostly all of us [1, 2]. There are OSN almost for everything, from cooperative work, car pooling, healthcare, friendship, love, affairs, healthcare, information, gaming, etc. In almost all of the cases, there are no two OSN that are built by the same software producer, and - quite often - mostly of them work on a competitive basis, and - for many different reasons (business, privacy, politics, etc.) - they are not open source and they are hosted by their private servers.

The interactions between those OSN is very little, since the Application Programming Interfaces (API) - while existing - are very weak and limited to accessing the buddy lists, walls, or accessing OSN X with the password of OSN Y. Even worst, the possibility of programming or at least interconnecting common features (e.g. accessing on different data-bases, link user names and passwords, chats, etc.) between different OSN is quite limited.

Beside of the “business needs” of any commercial OSNs, the myMed (<http://www.mymed.fr>) meta-social network represent a novel approach toward the natural interconnections of social applications. A short description of the framework can be found in the appendix of this paper. Technically speaking, myMed is an open source meta-framework providing software modules, some built-in templates,

INRIA Sophia-Antipolis Méditerranée, e-mail: Luigi.Liquori@inria.fr

a run-time cloud-based platform based on a customized version of Ubuntu OS, a Cassandra noSQL data-base, the well-known Apache and GlassFish web servers, a pretty clear installation docs, and a development environment that could be used to *build-install-run* OSN on the same hardware platform. Interestingly, the hardware platform could be a collection of standalone workstation PC's connected in a distributed ring topology instead of a classical set of racks containing skinned and powerful servers usually hosted in classical expensive data-center.

The main content of this paper are:

- introducing and justifying some technical choice underneath myMed from the implementor point of view;
- introducing and discussing some social and cooperative issues from the end-user point of view;
- list some open problems and questions.

The target of this paper is mostly for OSN implementor but also (in a minor mode) for OSN end-users and OSN researchers.

2 myMed's Technical Choices and Philosophy

2.1 *Distributed vs. Centralized*

The first question we asked when we started this project was about what kind of hardware should be bought: the first natural choice it would have been to fix a “data center” location, chosen between the consortium members, and buy a suitable number of rack fulfilled of skinned PC with minimal user interface, powerful CPU and huge hard disk. Quickly, we face up to the problem of the high cost to maintain during and after the project timeline this solution. Two main lines of expenses normally are induced by this choice: *i*) the presence of a well equipped room (air conditioned, sufficient power grid, etc.) and *ii*) the presence and the cost of an engineer maintaining the data-center: in few words, more than 20% of the entire contract budget it would have been devoted to the hardware.

Then, a second choice raised and it was finally chosen: buy 50 workstations HP Z400, equipped with Intel Xeon W3520 2.66 GHz, 8GB Ram and 2 HD of 1TB, 19" screens, US keyboards and mouses) and distribute it towards the eligible territories of Interreg Alcotra UE call <http://www.interreg-alcotra.org/2007-2013/?pg=territori>. The pros of this solutions are that we distribute the PC toward the territory (see Figure 1 left) and we install a customized version of Ubuntu OS, called UME for *Ubuntu myMed Edition*, that pragmatically split the PC into two distinct semantic areas: the first one playing the role of one cloud-node (see Figure 1 right) of the distributed data-base and the second one playing the role of a simple *desktop PC* equipped with an open source suite of a text editor, a spreadsheet, an internet browser, and other classical desktop applications. The cons of this solution are *i*) the need to cope with a general latency of the cloud, due to the fact

that the cloud ring was *geographically distributed*, hence a major care in choosing the installation sites where the internet connection was more than efficient, and *ii*) a major replication factor in the Cassandra noSQL data-base itself.

2.2 Design a Common Template for Build OSN

myMed is a social platform where many OSN can be conceived and hosted. A “template” has been designed with the help of a company specialized in HCI that standardize the most common OSN features, like buddy lists, reputation system, connexion with other commercial OSN and so on. Changing the layout of one myMed social application (called in jargon *sociapp*) is not strictly forbidden but strongly discouraged. This standardization make life simpler both to developers and to end-users. A *proto-store* containing all the sociapps developed and hosted within with the myMed store is available. The submission principle is quite similar to others stores, like e.g. the Apple Store[®] one, were a validation phase is required before the sociapp would be exposed on the store to myMed end-users: the difference here is in the fact that the sociapp is both exposed in the store, but, once installed, is also hosted on the myMed cloud. Actually the construction of sociapps demand standard skills in web programming languages, like Php, Javascript and Html. We plan, in the future releases of the template, to simplify as much as possible the process to build one sociapps by making it available to non experienced people.

2.3 Dealing with a Centralized Login and Tower of Extended Profiles

It is well-known that in all OSN, the end-user fill its profile that totally or partially is exposed in the network. Each OSN hosted by myMed have a proper user profile. Each end-user that want to enter in the myMed community should first enter into a simple registration step, either by filling an email and a password (a simple confirmation by mail is required to activate the myMed account) or by providing the Facebook, Twitter, or Google+ account (see Figure 2 left). This is quite common in many OSN and do not represent any shift in common practice of end-users. Once logged in the *myMed desktop* (see Figure 2 right), the end-user can access to the myMed store and browse the different OSN available. Each myMed OSN (or myMed sociapp) being available on the store should provide, as common practice in other stores, a clear explanation of their objectives and optionally some screen captures. The end-user can decide to install the sociapp on its desktop. Once the sociapp is installed on the desktop and launched, the end-user is asked to fill a, so called, *extended profile*, i.e. some extra informations that are necessary to use the OSN. Informations that are in common with already installed sociapps are automatically displayed while extra information must be filled in order to use the sociapp.

In other words, the myMed user profile grow each time new sociapps are installed: we call this feature a *tower of extended profiles*. Inversely, the end-user can decide to uninstall some previously installed sociapps: in this case all the extended profile and the data posted to the ex-user are deleted: this practice of deleting all personal datas and activity is enforced in the myMed modules of user deletion: they improve the myMed politics to do not store any data of deleted users.

2.4 Dealing with Different Browsers and Standalone Mobile Applications

The main philosophy of the myMed framework follows the *write less, do more* slogan. This objective was reached using extensively librairies like JQuery (<http://jquerymobile.com/>) that allow myMed sociapps to run responsively a on large list of browsers and mobile browsers running of fixe and mobile platforms. A nice shortcut was implemented to access directly to a given sociapp instead of accessing via the myMed desktop: the web address <http://www.mymed.fr/myOSN> allow to access directly to the login page of the myOSN sociapp. This shortcut is also useful if we want to package and distribute one sociapp via other stores for mobile devices, like the Apple or the Android ones. In that case only the application code of the packaged sociapp is submitted to the store. A bit of care should be done with iOS applications because *i)* the code must be wrapped into an Objective-C “mule” and *ii)* the code will be examined by Apple Store referees and rejected if the features of the iOS myMed sociapp are exactly the same than the pure web sociapp (normally we add some “mini-features” to obtain the Apple Store approval).

2.5 Dealing with Intra Sociapp Communication

All social applications share, as described in the introduction, a lot of software modules, like buddy list, geo-localization, interconnection with other commercial OSN, maps (via Google Map), etc. But, what about the *politics of data sharing inter myMed OSN*? The chosen and enforced solution in the myMed framework, was the one of deciding at the very first design phase level, what kind of informations should be kept private and what could be retrieved, via simple API, by others myMed OSN. From the developer point of view this feature in the myMed distribution represent a great advantage in simplifying the process of interconnecting OSN all built with the same meta-framework: we think it can also improve the *open data* movement, by opening as much as possible datas of a single myMed OSN to others myMed OSN. Last but not least remind that all the OSN use the same noSQL data-base whose access can be very fast.

2.6 Dealing with Asymmetric Viral Communication of Commercial OSN

The most used practice of interconnecting commercial OSN is based on a form of “asymmetric viral communication”.

Informally: asymmetric viral communication means that commercial OSN all share the same objectives: increase the number of members that, directly or indirectly, allows to make more money with it! To do that they usually “open” a bit their data-base, by offering suitable API for small read/write operations. Read operation includes: accessing to buddy list, reading some open information about the end-user profile, offering simple mechanism to allow login/logout to others OSN (this open, *de facto*, the room to crawl informations of others OSN since they share login and password!). Write operation includes: expressing an advice on some data present in another OSN, (the well-known Like, +1, Tweet, etc. buttons) that produce to literally copy the entire data on the wall in case the data reside on a dynamic web page, or just adding a reference pointer in case the data reside on a static web page. Unfortunately, the inverse operation, i.e. “pump” any datas from commercial OSN is quite often forbidden and this for a simple reason that could be resumed in the slogan *what is mine is mine what is yours is also mine* This asymmetry is dictated by simples business rules applied to the fact that OSN’s data-bases are a gold mine for their owners, and sharing information goes directly against the business rules and strategies of commercial OSN. *Open Data* strategies very often does not bring to a commercial model.

A bit more formally: asymmetric viral communication means that user A can relay (or post or publish) a duplicated copy or a pointer of a record published in OSN X into another OSN Y, provided user A have also an account on OSN Y, and *de facto* forbid in mostly cases the inverse operation, aka subscribe for a different user B on OSN Y to any publication of user A on OSN X even if user B does not have an account on OSN X. Because of this asymmetry, the well know paradigm of Publish/Subscribe [3] at the basis of many CSCW applications cannot be fully exploited inter OSN, leaving only their use intra OSN, the latter use being of less impact in coordination and co-operation.

3 Future work: Experimenting myMed on a Large Scale Testbed

We finish with a spare list of future work and some open questions.

- myMed sociapps (should be) easier to build and deploy. The patters is: start from the myMed Template; add or remove features, play with the design and the interface, implement the sociapp specification (i.e. what the sociapp really must do) and finally deploy on the myMed cloud, publish on the proto-store, and eventually package the code and submit to other commercial stores for mobile apps.

- myMed framework (should be) more extensible. Its modular architecture could be improved, since other developers (we can imagine an open source community) could easily install new modules and new sociapps.
- myMed framework can be considered as a distributed Social OS? It is installed on a distributed and geographically decentralized pool of servers (see Figure 1 left). Modifications of internal myMed modules would not affect the behavior of all sociapps using those modules. The project is actually hosted on a INRIA forge repository and the managed by Git, a distributed revision control and Source Code Management (SCM) system. One machine on the cloud is devoted to deploy semi-automatically the modifications pushed on the main repository.
- myMed framework (should) promotes more collaboration and cooperation between OSN. The sharing of all social modules have the positive effect of greatly facilitate OSN interconnection but more care in *security policies* of accessing personal extended profile could be done.
- myMed framework and myMed sociapps (should) be compatible with an ad hoc economical model? It is well know that economical models for commercial OSN are quite often related to advertising, or buying intra OSN features. The myMed interconnectivity by construction, open a way to novel business models, like “the more you open the more you earn”.
- myMed framework (should) feature an unique Human Computer Interface? A common template is provided to expert users that want to implement a proper OSN. Do we need to be compliant with some graphical chart or it would be better to leave more freedom in conceiving HCI?
- myMed framework can run on different instances. Can different myMed instances cooperate together? As in higher-order languages, the same cooperation level featured between different OSNs running on one myMed instance can be applied on different myMed instances running on different hardware. This would be subject of a further evolution of inter-cooperation and connection of myMed instances and their OSN running inside it. a lot of care must be given in building coherent “meta basic profiles” and “meta towers of extended profiles”.

Acknowledgements We warmly acknowledge Alain Giboin and the Reviewers for their useful suggestions and comments on the abstract of the paper, and Prof. Alessandro Andretta for careful proofreading.

References

1. J. Di Micco, D.R. Millen, W. Geyer, C. Dugan, B. Brownholtz, and M. Muller (2008). Motivation for Social Networking at Work. Proceedings of the 2008 ACM conference on Computer Supported Cooperative Work (CSCW), pages 711-720, ACM press.
2. H. Zhang, M. De Choudhury, and J. Grudin (2014). Creepy buy Inevitable? The Evolution of Social Networking. Proceedings of the 2014 ACM conference on Computer Supported Cooperative Work (CSCW), pages 368-378, ACM press.
3. P. Eugster, P. Felber, R. Guerraoui, and A-M Kermarrec (2003). The Many Faces of Publish/Subscribe. ACM Comput. Survey, volume 35, number 2, pages, 114-131, ACM press.

Appendix: myMed in a Nutshell

myMed is an experimental framework for building and hosting Open Social Networks. It was conceived as a common effort between five academic sites (Inria, Polytechnic of Turin, University of Turin, University of Piemonte Orientale, University of Nice Sophia Antipolis) and few local start-ups. myMed is an open source project, which facilitate and accelerate the development of *ad hoc* social applications (called in myMed jargon “sociapps”) running over an heterogeneous “Plateau” of platforms, such as PCs, Smartphones and Tablets running iOS and Android. myMed provides a rich framework for publishing, searching and subscribing to content: the engine is built on top of a distributed noSQL database. In its current version, it provides high scalability and fault tolerance. The myMed framework allows you to easily build social web applications: it features geolocalization, points of interest in charts, buddy lists, profile management, content/user reputation, built-in cooperation and coordination among different OSN running on it, proto OSN store, etc. In a nutshell the myMed framework is composed by:

- a Software Development Kit (SDK) to develop fixed and mobile web sociapps, running on many Web browsers but also natively on Smartphones and Tablets equipped with Android or iOS. Sociapps, by their name, must have a strong social flavors (open social networks, closed social networks, enter-prise social networks, micro/nano social networks and so on). Thanks to the rich, general-purpose, catalog of modules in the framework, every module can be freely used without interfering with other sociapps, in a true “Lego” fashion. The program is distributed under the Apache V2 free license. The TTM (“Time To Market”) envisaged to develop a sociapp using the myMed SDK can be estimated from 1 to 3 months employing 1 or 2 junior web programmers.
- A “cloud” to execute the “sociapps” represented by a “backbone” of 50PCs, distributed through the “AlpMed” EuroRegion following some precise efficiency criteria (as example the presence of Internet running on optical fibers). Part of those PCs have a double function:
 - ensure the good behaviors of all the running sociapps, and
 - offer services other than those offered by myMed, such as a web browser, an open-source Office suite, a private disk of little size, logically separated from the noSQL space that can be used by others users (we call this “an elastic usage of the myMed cloud PC”).

Those PCs can be accessed via a private login and password generated by the framework on demand. The operating systems running on those PC (Ubuntu myMed Edition, UME) is also open source and it is based on a customized version of the Ubuntu operating system. To guarantee the quality of the execution of the sociapps, we require that all machines belonging to the backbone are constantly running (on state).

- A little collection of “proof of concept” sociapps to validate, experiment, and testing the development kit and the execution cloud. These sociapps have been

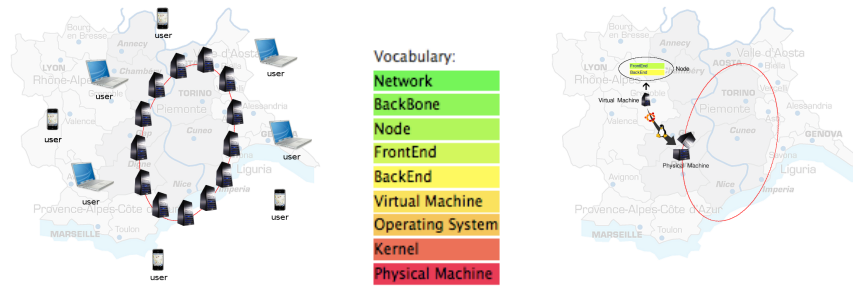


Fig. 1 The myMed backend

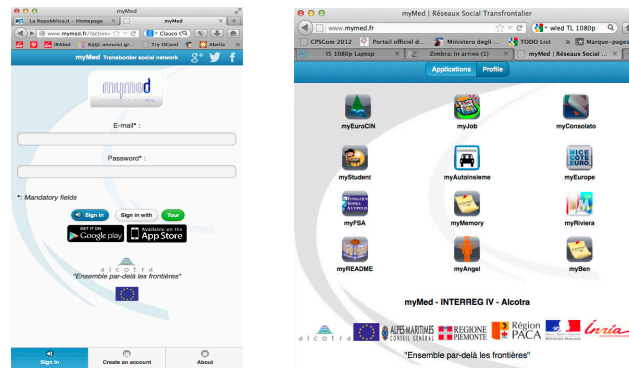


Fig. 2 The myMed frontend

conceived with the precious help of the “Civil society” of the EuroRegion “AlpMed” (States, Regions, Prefectures, Associations, Chambers of Commerce, Municipalities, Universities, etc.) that have played a role of “maître d’oeuvre” (or experts) in a given “application domain”. The quasi totality of the sociapps are available on the myMed web platform at the address <http://www.mymed.fr/?action=login> but also on the most common mobile application stores, such as the Apple Store and the Google Play Store Markets. Figure 1 and 2 shows the backend and the frontend of the myMed system.